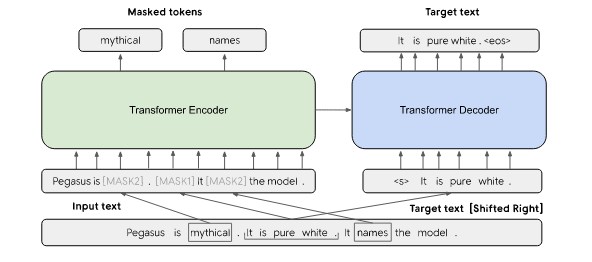
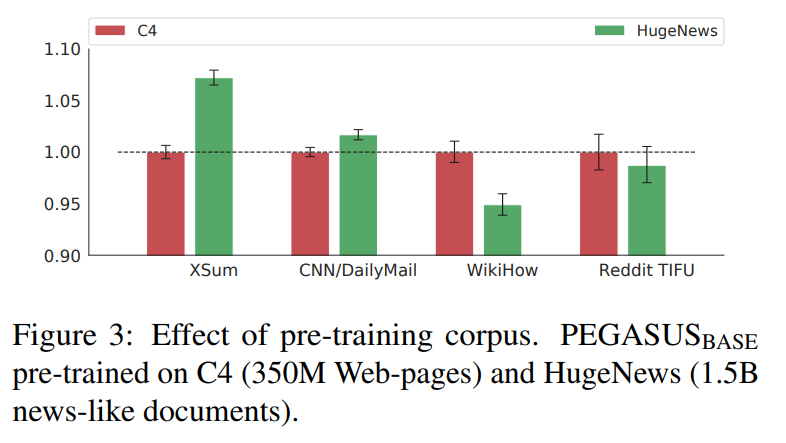
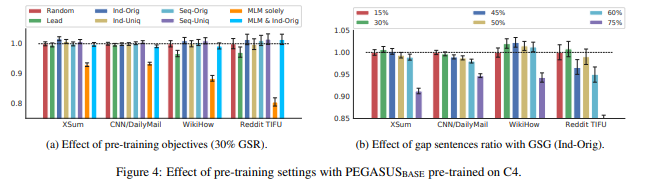
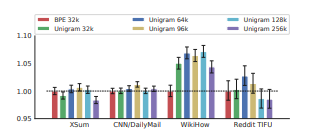
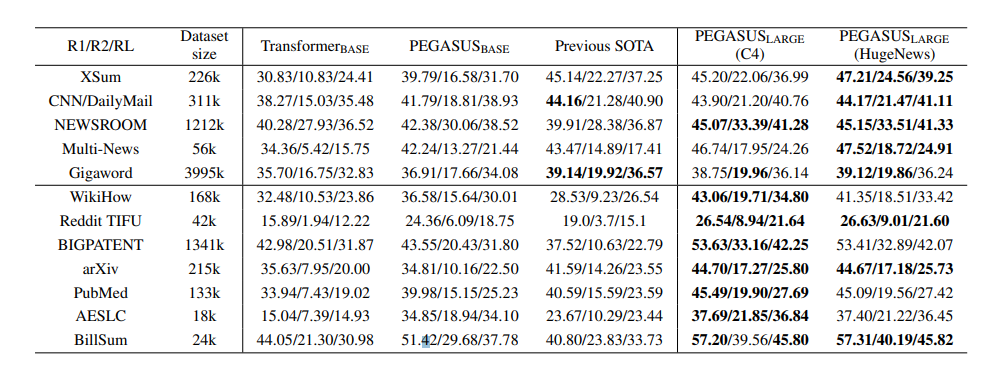
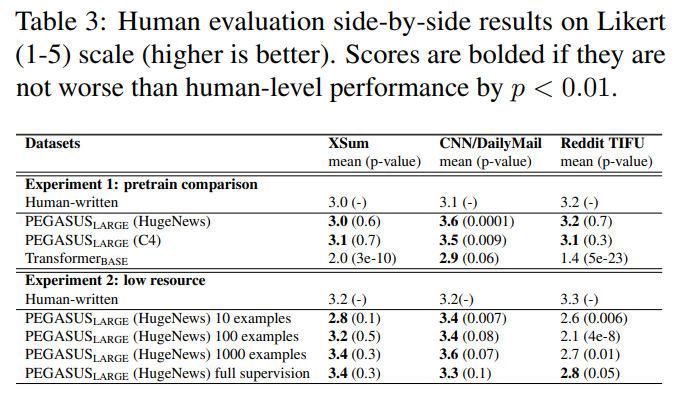
PEGASUS : Pre-training with Extracted Gap-sentences for Abstractive Summarization(2020) 리뷰

1. Introduction

* 단순히 입력으로부터 문장을 추출하는 Extractive summarization과 달리, abstractive summarization은 상대적으로 어렵지만 다양한 표현을 생성할 수 있음. Transformer 기반 seq-to-seq 모델이 좋은 성능을 보이고 있음
* 본 논문에서는 문서 내 단어가 아닌 전체 문장을 마스킹하고, 이러한 Gap sentences 을 문서의 나머지 부분으로부터 생성하는 것을 summarization을 위한 pretraining objective로 제안함(Gap Sentences Generation, GSG). GSG를 사용하여 거대 코퍼스에서 Transformer encoder-decoder를 pretraining 한 모델이 PEGASUS.
* XSum, CNN/DailyMail 등의 데이터에서 인간 수준의 요약 성능을 입증했으며 SOTA 달성

2. Pretraining Objectives  


* + Gap Sentences Generation(GSG)
    - BERT의 MLM 방식과 같이 general한 pretraining obejctive가 아니라, abstractive text summarization task의 성능을 높일 수 있는 pretraining objective를 착안
    - Pretraining objective가 downstream task(summarization)와 유사할수록 fine-tuning 했을 시, 모델의 성능이 높아질 것으로 가정 -> GSG는 토큰이 아닌 일부 sentence를 Masking하여 이를 맞추는 방식으로 학습하게 되고 이 과정이 summarization 방식과 유사함
    - 따라서, Transformer 모델이 맞춰야 하는 Gap sentence가 중요할수록 summarization 방식과 유사함
  + Masked Language Model(MLM)
    - 기존 BERT에서는 input text의 15%의 token을 선택하여, 그 중 80%는 mask token으로 변환하고, 10%는 random token, 나머지 10%는 그대로 사용
    - 위 PEGASUS 모델의 아키텍처를 보면 GSG와 MLM이 동시에 적용되고 있으나, 실험 결과   
      MLM이 downstream task의 성능 향상에 영향을 주지 않아 최종 모델에서는 MLM을 포함하지 않음
  + Datasets
    - Pre-training을 위해 사용된 corpus는 C4, HugeNews
      * C4(Colossal and Cleaned version of Common Crawl) : consist of text from 350M web-pages(750GB)
      * HugeNews : a dataset of 1.5B articles (3.8TB) collected from news and news-like websites from 2013-2019
    - Downstream tasks/datasets
      * Public datasets인 Tensorflow Summarization Datasets 12개 사용(XSum, CNN/DailyMail, arXiv 등)
  + Experiments
    - 학습 모델 종류
      * PEGASUS-base
        + number of layers of encoder and decoder(L) : 12
        + hidden size(H) : 768
        + feed-forward layer size(F) : 3,072
        + number of self-attention heads(A) : 12
      * PEGASUS-large
        + number of layers of encoder and decoder(L) : 16
        + hidden size(H) : 1024
        + feed-forward layer size(F) : 4,096
        + number of self-attention heads(A) : 16
    - Effect of pre-training corpus  
      
      * Pre-training corpus의 종류에 따라 downstream task의 성능에 영향을 줌
      * HugeNews 기반으로 PEGASUS-base 모델은 news 기반 데이터셋(XSum, CNN/DailyMail)에서 높은 성능을 기록한 반면, non-news 기반 데이터셋(WikiHow, Reddit TIFU)에서는 낮은 성능을 기록
    - Effect of pre-training objectives  
      
      * GSR(Gap Sentences Ratio)
        + 각 데이터에 대해 실험 결과 30% 선택(오른쪽 그림)
      * Pre-training objectivts(각 조합별 실험 진행)
        + Random : 랜덤하게 m개 문장 Masking
        + Lead : 첫 m개 문장 Masking
        + Ind : 특정 문장, 남은 문장들 간의 Rouge1-F1를 독립적으로 계산하여 높은 m개 문장 Masking
        + Seq : 문장간의 조합을 고려하여 선택한 조합과 남은 문장들 간의 Rouge1-F1를 계산하여 높은 조합 문장들 Masking
        + Uniq : 여러 단어가 중복인 경우 하나로 counting 하는 방식
        + Orig : 중복을 모두 counting 하는 방식
        + 논문에서는 Ind-Orig 성능이 가장 좋았으며 그 다음으로 Seq-Uniq의 성능이 높게 나옴
    - Effect of vocabulary  
      
      * Byte-pair encoding(BPE)과 SentencePiece Unigram간 교를 통해 최종적으로 SentencePiece Unigram을 사용하고 vocabulary size는 96k로 선정
    - Larger Model  
      
      * PEGASUS-large 모델의 경우 이전 SOTA모델 대비 12개 downstream tasks에서 모두 SOTA를 달성함
    - Human evaluation   
      
      * 실제 사람이 만든 summary와 PEGASUS-large모델이 만든 summary를 비교한결과, Reddit TIFU 데이터셋을 제외한 XSum, CNN/DailyMail 데이터셋에서는 PEGASUS-large모델이 만든 요약본이 사람이 만든 요약본보다 더 높은 성능을 나타냄
      * 사람이 만든 과정이 객관적으로 이루어진 결과인지에 대한 언급은 없음..